# DUALMIND: WHEN TO ANSWER AND WHEN TO CLARIFY IN TEXT-TO-SQL

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Text-to-SQL models often produce syntactically valid yet semantically incorrect queries, as likelihood-based decoding can silently collapse schema ambiguity without verifying user intent. We present DualMind, a neuro-symbolic framework that decouples probabilistic calibrated grounding from formal semantic consensus. It retains plausible interpretations via conformal calibration, refines candidates with certified external evidence, and emits SQL only when all survivors belong to a single semantic equivalence class. We prove a compositional error bound for adaptive multi-round clarification with no per-round error accumulation, derive a verifiable criterion for resolving semantic ambiguity, and develop an optimal questioning policy under a finite clarification budget. In a controlled synthetic grounding study with a 41,344-parameter causal Transformer and three seeds, allowing at most two simulated authoritative clarification answers increases output coverage to 96.5% while maintaining 6.2% selective error, demonstrating that DualMind can substantially resolve semantic ambiguity while preserving reliability among accepted queries.

## 1 INTRODUCTION

Consider the request "report the amount actually paid this month." An enterprise database contains both `orders.billed_cents` and `payments.paid_cents`. Selecting the former can produce valid SQL and a plausible number while answering the wrong question. A second choice concerns time: should "this month" refer to the order date or the payment date? A foreign-key check cannot settle either business interpretation. Such schema-grounding errors motivate our work: the gap is between a likely interpretation and an interpretation supported by the available evidence.

Transformer models assign probabilities to token sequences (Vaswani et al., 2017). Their predictive objective does not impose a proof obligation for each selected schema entity. This does not imply that Transformers cannot reason. Rather, model preference, grammatical validity, and semantic justification are distinct. Schema linking (Wang et al., 2020), constrained decoding (Scholak et al., 2021), and decomposed generation (Pourreza & Rafiei, 2023) address important parts of the problem, yet ambiguous questions may still admit multiple plausible SQL interpretations (Bhaskar et al., 2023). Selecting the highest-scoring valid candidate can silently collapse this ambiguity.

**Our approach.** DualMind assigns complementary responsibilities to two components. The probabilistic component represents plausible groundings and ranks useful clarification actions. The symbolic component checks explicit evidence, groups groundings by certified query semantics, and determines whether a query may be emitted. The components share a finite set of possible interpretations. Model scores determine which candidates calibration retains and guide question selection. Certified evidence and semantic agreement determine whether a query can be emitted.

The distinction between interpretations and answers is important. Two groundings can differ in an attribute irrelevant to a particular query and still compile to equivalent SQL. Requiring a unique grounding then asks unnecessary questions. Conversely, several structurally legal candidates can return different results on some legal database, even if they coincide on the current database. DualMind therefore resolves *semantic classes*, rather than token strings or every latent attribute.

**Contributions.** (i) We formulate calibrated grounding, certified evidence refinement, and semantic consensus in one interface, with an explicit separation between statistical and formal guarantees.



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

(ii) We prove a compositional, adaptive-refinement error bound, characterize semantic identification through separating questions, and derive a model-based optimal policy for a finite clarification budget. (iii) We instantiate the framework in a restricted SQL language and provide a reproducible study, independent SQL checks, and controlled assumption violations. Sections 2–4 define and analyze this rule, Section 5 evaluates its tradeoffs, and Section 6 positions it against prior work.

## 2 Problem formulation and correctness boundaries

An input $x$ consists of a request and a schema catalog. Let $\mathcal{H}(x)$ be a finite universe of interpretations. An interpretation $h$ specifies the choices needed by a compiler, such as a source table, a value field, a date field, a filter, and an aggregation. The intended interpretation is $h^\star$. A trained model provides $p_\theta(h \mid x)$. The implementation uses one generated plan token to represent $h$. The framework also permits probabilities induced by longer generated sequences.

**Query semantics.** The compiler $c_x : \mathcal{H}(x) \to \mathcal{Q}$ produces SQL in a declared fragment. Let $\mathcal{D}(x)$ be the legal database instances, including type, key, and numerical restrictions. Query equivalence is

$$q \equiv_x q' \quad \Longleftrightarrow \quad \forall D \in \mathcal{D}(x),\ [\![q]\!]_D = [\![q']\!]_D. \tag{1}$$

A semantic signature $\sigma_x(h)$ is *sound* if equal signatures imply equivalent compiled queries. It is *exact* if the converse also holds. This is an interface requirement, not an assumption that arbitrary SQL equivalence is easy. Our fragment admits a simple exact signature, proved in Appendix B. A conservative proof system may instead merge only equivalences it can establish. Unknown equivalences remain separate.

**Evidence and trust.** Certified evidence is a predicate $e(h) \in \{0, 1\}$ with a provenance outside the model's unverified preference: for example, a curated business definition or an explicit user answer. Its feasible set is $E = \{h : e(h) = 1\}$. Structural facts such as a foreign key are not substitutes for an intent assertion such as "use received payment amount." Evidence is *truth-preserving* on an interaction when it never removes $h^\star$. We permit a failure event of probability at most $\delta$, covering incorrect metadata or answers. A fluent model-generated rationale is not certified evidence.

**Scope of correctness.** Structural validity establishes that a query uses legal schema objects. Evidence consistency establishes agreement with encoded requirements. Intent correctness additionally requires that those requirements and the candidate universe faithfully represent the request. Our guarantee is conditional on this representation interface. In particular, no amount of deduction from two identically typed amount fields can determine their business meaning when the catalog and request do not distinguish them.

**Coverage and error metrics.** For an output $\widehat{q} \in \mathcal{Q} \cup \{\bot\}$, where $\bot$ denotes abstention, define

$$\begin{aligned} \kappa &= \mathbb{P}(\widehat{q} \neq \bot), \qquad r = \mathbb{P}(\widehat{q} \neq \bot, \widehat{q} \not\equiv_x c_x(h^\star)), \\ r_{\mathrm{sel}} &= \mathbb{P}(\widehat{q} \not\equiv_x c_x(h^\star) \mid \widehat{q} \neq \bot) = r/\kappa \quad (\kappa > 0). \end{aligned} \tag{2}$$

Here $\kappa$ is output coverage, $r$ is the wrong-output rate over all inputs, and $r_{\mathrm{sel}}$ is the error rate among accepted queries. We report coverage and both error rates. A low $r$ alone can be achieved by abstaining on everything. Moreover, a bound on $r$ is not the same as a bound on $r_{\mathrm{sel}}$.

## 3 The DualMind framework

Figure 1 separates two responsibilities. In panel (a), the Transformer scores schema interpretations, and calibration retains a plausible set $C_\alpha(x)$. A high score expresses preference. It does not certify that a table, amount field, or date field matches the request. In panel (b), certified evidence intersects that set, and semantic signatures map survivors to query meanings. Agreement among those meanings controls emission. Neural weights guide clarification.

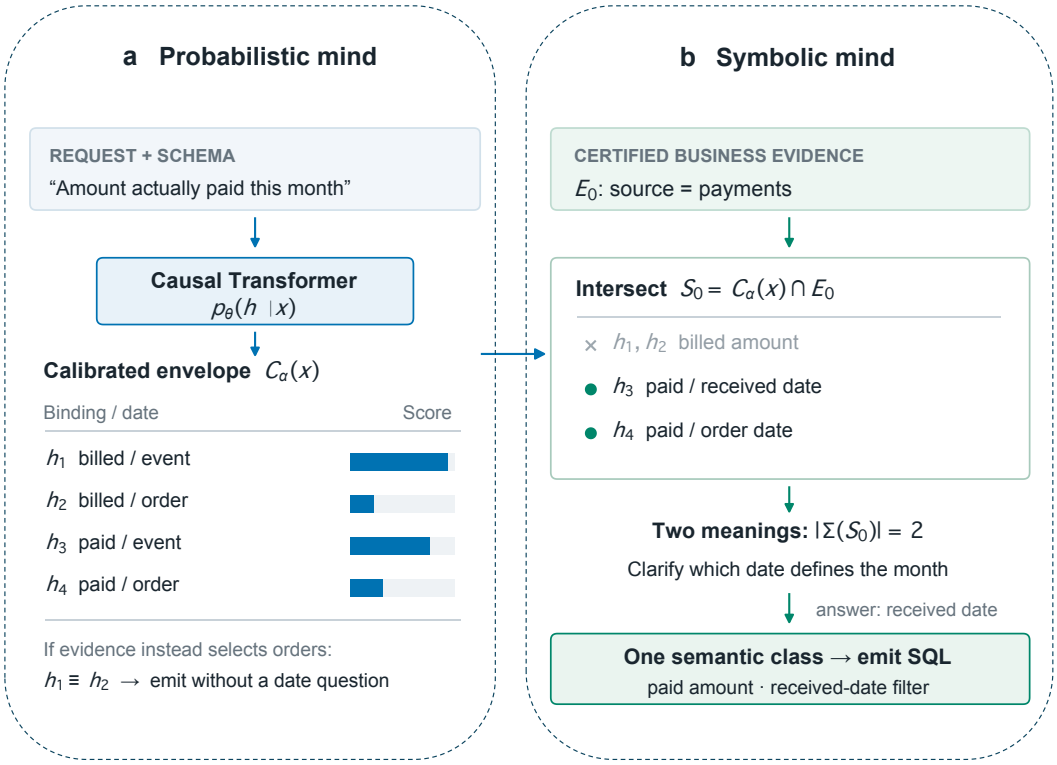


Figure 1: DualMind illustrated on amount and date grounding. **(a)** A probabilistic model supplies a calibrated interpretation set (score bars are schematic, not experimental measurements). **(b)** Certified evidence removes incompatible bindings. A date clarification resolves the remaining semantic disagreement. If evidence instead selects orders, equivalent date bindings permit emission without that question. Empty sets lead to abstention.

### 3.1 Calibration, evidence refinement, and semantic consensus

After fitting the neural model on a separate training set, use $n$ calibration examples $(X_i, H_i^\star)$ and scores $s(x, h) = 1 - p_\theta(h \mid x)$. For $0 < \alpha < 1$, set

$$k = \lceil (n+1)(1-\alpha) \rceil, \quad \widehat{t} = s_{(k)}, \quad C_\alpha(x) = \{h \in \mathcal{H}(x) : s(x,h) \le \widehat{t}\}, \tag{3}$$

where $s_{(k)}$ is the $k$th calibration order statistic and $\widehat{t} = +\infty$ if $k = n+1$. This is a standard split-conformal construction. We do not propose a new conformity score. Its rank argument gives marginal coverage under exchangeability (Romano et al., 2020). The finite universe must cover the modeled ground-truth labels. An uncalibrated retrieval stage that removes labels is an additional approximation, not part of this guarantee.

**A grounding example (Figure 1).** The four candidates combine billed versus paid amount with event versus order date. Aggregation and status filtering are fixed. Evidence that the request concerns payments removes $h_1$ and $h_2$, but leaves two meanings: filtering payments by receipt date ($h_3$) or by the associated order date ($h_4$). Because orders and payments can fall in different months, the system asks which date defines the period. The answer “received date” selects one semantic class and permits emission. If evidence instead selects orders, $h_1$ and $h_2$ remain distinct interpretations but have identical semantics because an order’s event date is its order date. No date question is needed. This distinction between candidate uniqueness and semantic agreement motivates the stopping rule. All decisions are relative to the retained set and cannot repair an initial omission of the intended interpretation.

Let $E_0$ encode initial certified evidence. The retained set is $S_0 = C_\alpha(x) \cap E_0$. For subsequent answers $e_1, \ldots, e_t$, maintain

$$S_t = C_\alpha(x) \cap E_0 \cap \bigcap_{j=1}^{t} E_j, \qquad \Sigma(S_t) = \{\sigma_x(h) : h \in S_t\}. \tag{4}$$

DualMind emits $c_x(\widehat{h})$, for any $\widehat{h} \in S_t$, exactly when $S_t \neq \varnothing$ and $|\Sigma(S_t)| = 1$. It abstains on an empty set: inconsistency is not unanimous agreement. If multiple classes remain, it asks a permitted clarification when budget is available. Otherwise it abstains. Choosing the highest-scoring member within one certified class does not change the answer semantics.

The order of operations matters. Conditioning probabilities on evidence and then selecting an arbitrary high-confidence singleton is not Eq. (4). Likewise, discarding low-probability members after calibration changes the envelope. DualMind can rescore survivors for planning, but keeps the original membership condition for its guarantee.

### 3.2 Budgeted clarification over semantic classes

A question $a$ has a finite answer function $g_a : \mathcal{H}(x) \to \mathcal{O}_a$. Its truthful answer is $g_a(h^\star)$. It partitions $S$ into $S_{a,o} = \{h \in S : g_a(h) = o\}$. Examples ask whether the intended amount is billed or received, or which date defines the reporting period. Questions with one nonempty child provide no information and are omitted.

For efficiency planning only, normalize the fixed model weights on $S$:

$$w_S(h) = \frac{p_\theta(h \mid x)}{\sum_{u \in S} p_\theta(u \mid x)}, \qquad w_S(S_{a,o}) = \sum_{h \in S_{a,o}} w_S(h). \tag{5}$$

Assume positive mass on every retained interpretation. A full-support smoothing rule is also possible. Define $V_b(S)$, the maximum *model-weighted* probability of resolving the semantics within $b$ answers:

$$V_b(S) = \begin{cases} 0, & S = \varnothing, \\ 1, & |\Sigma(S)| = 1, \\ 0, & b = 0 \text{ and } |\Sigma(S)| > 1, \\ \max\limits_{a} \sum\limits_{o : S_{a,o} \neq \varnothing} w_S(S_{a,o}) V_{b-1}(S_{a,o}), & \text{otherwise.} \end{cases} \tag{6}$$

The maximum over no informative questions is zero. DualMind chooses a maximizing question, receives the answer, and recurses on the surviving set. Memoization makes this exact policy inexpensive for our 16-world instance. Its objective is coverage under model weights, not guaranteed real-world utility. Inaccurate probabilities can degrade efficiency without invalidating truth-preserving set refinement.

For larger universes, a semantic edge heuristic is an alternative. Let $W_S(T) = \sum_{h,u \in T} w_S(h) w_S(u) \mathbf{1}\{\sigma_x(h) \neq \sigma_x(u)\}$. The heuristic maximizes expected removed cross-class mass, $\sum_o w_S(S_{a,o})[W_S(S) - W_S(S_{a,o})]$. This is motivated by equivalence-class active learning (Golovin et al., 2010). We do not claim a new approximation ratio for the implementation. World-entropy and random-question policies provide additional controls.

**Decision procedure.** Compute $S_0$ once. At each step, first check for an empty set or one semantic class. If unresolved and budget remains, select a question with Eq. (6), intersect with the answer cell, and decrement the budget. Emit only on semantic consensus. In our fragment, a certificate consists of the retained world IDs, evidence assignments, and their common normal form. A small enumerating checker can replay this decision. It certifies agreement inside the envelope, not truth of the business dictionary.

## 4 Theoretical properties and limits

We separate deterministic properties of a retained set from statistical coverage of the intended interpretation. We establish when consensus suffices, bound errors under adaptive refinement, and

characterize which questions resolve ambiguity and how to choose them. Full proofs appear in Appendix A.

**Proposition 1** (Sound consensus and its exact boundary)**.** *If $h^\star \in S$ and $\sigma_x$ is sound, emitting on semantic consensus yields a query equivalent to $c_x(h^\star)$. If $\sigma_x$ is exact, consensus is also necessary for any single-query rule that must be correct for every $h \in S$. Thus it accepts every set on which a universally correct single answer is possible.*

*Proof.* All retained interpretations have the emitted signature, so soundness gives equivalence to the intended query. Conversely, if two members compile to inequivalent queries, no query can be equivalent to both, by transitivity of equivalence. Exactness identifies precisely these incompatible sets. □

This proposition is an interface fact, not by itself the main novelty. It explains why unique-grounding abstention is unnecessarily conservative and why likelihood maximization cannot establish evidence-based certainty. With an incomplete signature, emission remains sound, but the necessity statement no longer applies.

**Theorem 2** (One-envelope guarantee under adaptive refinement)**.** *Suppose $\mathbb{P}(h^\star \notin C(X)) \le \eta$. Let $G$ be the event that every evidence restriction along the actual, possibly adaptive interaction preserves $h^\star$, with $\mathbb{P}(G^c) \le \delta$. Let the compiler and signature be sound on an event $K$ with $\mathbb{P}(K^c) \le \epsilon$. Any stopping rule that emits only on nonempty consensus in subsets formed by Eq. (4) satisfies*

$$r \le \min\{1, \eta + \delta + \epsilon\}, \qquad r_{\mathrm{sel}} \le \min\{1, (\eta + \delta + \epsilon)/\kappa\}. \tag{7}$$

*For Eq. (3), fixed training, and exchangeable calibration and test labels, $\eta \le \alpha$. The bound has no additional factor for the number of truth-preserving clarification rounds.*

*Proof.* On $\{h^\star \in C(X)\} \cap G \cap K$, every visited $S_t$ contains $h^\star$. Proposition 1 applies whenever the system emits, regardless of the stopping time. A wrong emission is therefore contained in the union of the three failure events. Apply the union bound and divide by $\kappa$ for conditional error. The conformal statement follows from the exchangeable-rank argument, including ties conservatively. □

The adaptive statement is useful because clarifications are selected using the model and earlier answers. It does not require a separately calibrated model for every interaction state: the original envelope is retained. However, $\delta$ is a failure probability for the *whole interaction*. If only per-answer error bounds are known, their accumulation must be included in $\delta$. Nor does Eq. (7) promise an error rate of $\alpha$ among accepted queries. At low coverage that interpretation can be substantially wrong.

**Theorem 3** (Semantic identification certificate)**.** *For exact signatures, define the disagreement pairs $U(S) = \{\{h, u\} \subseteq S : \sigma_x(h) \ne \sigma_x(u)\}$. A fixed collection of questions $A$ resolves semantics for every possible truthful answer transcript if and only if*

$$\forall \{h, u\} \in U(S), \quad \exists a \in A : \; g_a(h) \ne g_a(u). \tag{8}$$

*Consequently, if two disagreeing interpretations have identical answers to every available question, no adaptive policy using only these questions can be uniformly correct and always emit. Conversely, asking all questions in a separating collection suffices.*

*Proof.* Each complete answer transcript defines a cell. That cell is semantically homogeneous exactly when it contains no disagreement pair. A pair belongs to the same cell exactly when all question answers coincide, yielding Eq. (8). Indistinguishable pairs follow the same adaptive path. Proposition 1 rules out a universally correct single output on that path. □

This supplies a checkable criterion for whether a business dictionary or clarification interface contains enough information. It also turns minimum-cost nonadaptive clarification into a weighted set-cover problem over disagreement pairs. We use the characterization, not a claim that this classical combinatorial problem has become easy. In particular, questions that separate aliases but no query semantics are unnecessary.

**Theorem 4** (Finite-budget model-based optimality)**.** *With deterministic truthful answer functions, fixed positive weights, unit question costs, and budget $b$, Eq. (6) equals the greatest probability, under $w_S$, that any adaptive policy reaches a homogeneous nonempty cell within $b$ questions. Choosing a maximizing action recursively attains this value.*

*Proof.* The boundary cases are forced. For an unresolved state, every policy first chooses a question. Conditioning on its answer gives the weighted value of a policy with one fewer question. Induction bounds each child by $V_{b-1}$, and selecting maximizing child policies attains the bound. Randomizing the first question cannot exceed a maximum of deterministic choices. □

This is Bellman's principle applied to the semantic stopping condition. The model need not be a correct posterior for Theorem 2. It would need to be one to interpret Theorem 4 as true-distribution optimality. With $N$ worlds, $A$ questions, and at most $O$ answers per question, naive memoization over subsets uses at most $(b+1)2^N$ values and $O(b2^N AON)$ work. We make no polynomial-time claim for arbitrary schemas. In our four-attribute cube there are at most $3^4$ reachable answer-restricted states for a fixed initial envelope.

**A truncation counterexample.** Suppose $C = \{h_1, h_2\}$, the two interpretations disagree, and the true interpretation is $h_2$. Top-1 truncation to $\{h_1\}$ creates false consensus. If arbitrary extra pruning removes the truth with probability at most $\beta$, the same proof gives the valid but weaker bound $r \leq \eta + \delta + \epsilon + \beta$. Without controlling $\beta$, calibration does not justify that pruning. This motivates checking the full small envelope in our implementation.

## 5 SQL INSTANTIATION AND CONTROLLED EVALUATION

We evaluate whether semantic consensus improves useful coverage, how clarification changes the coverage–error tradeoff, and where the assumptions fail. The study uses a small SQL language so that semantic equivalence can be proved and compiler behavior checked independently.

### 5.1 SQL FRAGMENT AND SEMANTIC EQUIVALENCE

We use `orders(id, customer_id, ordered_day, billed_cents, status)` and `payments(id, order_id, received_day, paid_cents, status)`. All fields are non-null integers. `order_id` references a unique order. One order may have multiple payments. Interpretations are four bits $h = (m, t, s, a)$: order versus payment source, event versus order date, all versus completed records, and sum versus count. Dates are filtered to $[30, 60)$. Status $1$ denotes completion. Payment queries use the many-to-one foreign-key join, which preserves payment multiplicity. Sum returns zero on an empty selection using `COALESCE`. Legal instances exclude integer overflow.

The exact signature is

$$\sigma(m, t, s, a) = (m, mt, s, a). \tag{9}$$

For order-source queries, event date already is order date, so $t$ is irrelevant. For payment queries the dates can differ. Thus 16 interpretations yield 12 semantic classes. This equivalence is proved over all legal instances, not inferred from one database. An independent Python row interpreter checks the SQL compiler on 256 generated databases, including an empty database, giving 4,096 matching executions. The tests additionally separate all unequal signature classes on this suite. The universal equivalence argument is separate.

### 5.2 PROTOCOL AND BASELINES

Each seed independently generates 4,096 training, 1,024 calibration, and 2,048 test examples. A world is sampled uniformly. A templated request expresses each of the four attributes using one of three synonyms, in random slot order. Each clue is independently omitted with probability $0.15$. A present clue is flipped with probability $0.03$. This creates missing or misleading linguistic clues. Initial certified evidence independently reveals each true attribute with probability $0.35$. Reveals represent idealized curated intent/catalog information. They are not inferred by the tested model.

Table 1: Controlled grounding results, mean ± sample standard deviation over three seeds (2,048 test cases each). Coverage, wrong-output rate, and selective error are percentages. Wrong-output rate uses all inputs. Selective error uses accepted inputs. Answers are averaged over all inputs. Interactive rows receive additional oracle information.

| Method | Coverage | Wrong outputs | Selective error | Answers |
|---|---|---|---|---|
| Neural argmax / structural check | $100.0 \pm 0.0$ | $30.8 \pm 1.6$ | $30.8 \pm 1.6$ | 0.000 |
| Confidence $\geq 0.9$ | $16.8 \pm 2.3$ | $1.7 \pm 0.5$ | $9.9 \pm 2.4$ | 0.000 |
| Contract-filtered argmax | $100.0 \pm 0.0$ | $20.8 \pm 1.5$ | $20.8 \pm 1.5$ | 0.000 |
| Calibrated semantic consensus | $37.6 \pm 11.5$ | $4.1 \pm 1.1$ | $10.9 \pm 0.7$ | 0.000 |
| DualMind, unique grounding only | $50.6 \pm 9.8$ | $3.5 \pm 1.3$ | $6.8 \pm 1.4$ | 0.000 |
| Full-support semantic consensus | $2.9 \pm 0.1$ | $0.0 \pm 0.0$ | $0.0 \pm 0.0$ | 0.000 |
| DualMind, no clarification | $54.7 \pm 9.9$ | $3.9 \pm 1.3$ | $7.0 \pm 1.4$ | 0.000 |
| DualMind, at most one answer | $91.9 \pm 2.9$ | $5.7 \pm 1.1$ | $6.2 \pm 1.0$ | 0.419 |
| DualMind, at most two answers | $96.5 \pm 0.2$ | $6.0 \pm 1.1$ | $6.2 \pm 1.1$ | 0.482 |

Clarifications return the true bit through a simulated oracle, so user effort is measured as answer count, not interaction time.

The causal Transformer has two layers, width 48, four heads, feed-forward width 96, and 41,344 parameters. It predicts one of 16 plan tokens from the templated request, after which the compiler selects concrete tables and columns. We train for 12 epochs using AdamW, learning rate $10^{-3}$, batch size 128, and no dropout, on CPU with four threads. Three fixed seeds are 11, 23, and 37. No test-based checkpoint selection is used. This is a closed-world plan-grounding experiment, not an evaluation of pretrained LLMs, learned open-schema linking, or general SQL generation. Distinct test examples may share grammar patterns, as expected for the stated synthetic distribution.

All methods use the same predictions and, when applicable, the same certified evidence. Neural argmax and a structural check coincide because every plan compiles. Contract-filtered argmax chooses the most likely evidence-consistent plan. Calibrated semantic consensus omits certified evidence. Unique-grounding DualMind requires $|S| = 1$, rather than one semantic class. Full-support consensus starts from all 16 worlds and gives a deterministic reference point. We also compare a fixed confidence threshold of $0.9$. Main DualMind uses $\alpha = 0.1$. Interactive comparisons have the same maximum answer budget. Gains from adding answers should not be attributed to architecture alone.

### 5.3 Results: useful coverage with explicit error accounting

Table 1 quantifies the central tradeoff. Figure 2(a) separates correct outputs, wrong outputs, and abstentions. Neural argmax produces a query on every input, but 30.8% of inputs receive the wrong semantics. With the same model, noninteractive DualMind achieves 54.7% coverage and 3.9% wrong outputs, corresponding to 7.0% selective error. This is a selective decision rule, so the error reduction is not a claim of improvement at matched coverage. Full-support consensus makes no wrong outputs in the study but covers only 2.9%, showing the cost of retaining every logically possible interpretation.

**Evidence and semantic grouping.** Two comparisons isolate the effects of evidence and semantic grouping. First, explicit evidence improves coverage over calibrated consensus without that evidence. Second, quotienting by query semantics adds 4.0 percentage points of coverage over requiring a unique grounding. The latter does not mean that every extra accepted case is empirically correct: the true world may already have been excluded by calibration. It means the extra cases are resolvable relative to the retained envelope, precisely the distinction in Proposition 1 and Theorem 2.

**Clarification and calibration.** Figure 2(b) plots clarification budget directly against coverage $\kappa$ and accepted-query accuracy $a_{\text{sel}} = 1 - r/\kappa$. With at most one authoritative answer, coverage rises to 91.9%. With at most two it reaches 96.5%. Over the same path, selective error falls from 7.0% to 6.2%, so $a_{\text{sel}}$ rises from 93.0% to 93.8%. The all-input wrong-output rate still rises from 3.9% to 6.0% because the system emits on far more inputs. This increase does not mean that accepted queries become less accurate. The simulated answers reveal ground-truth attributes that were previ-

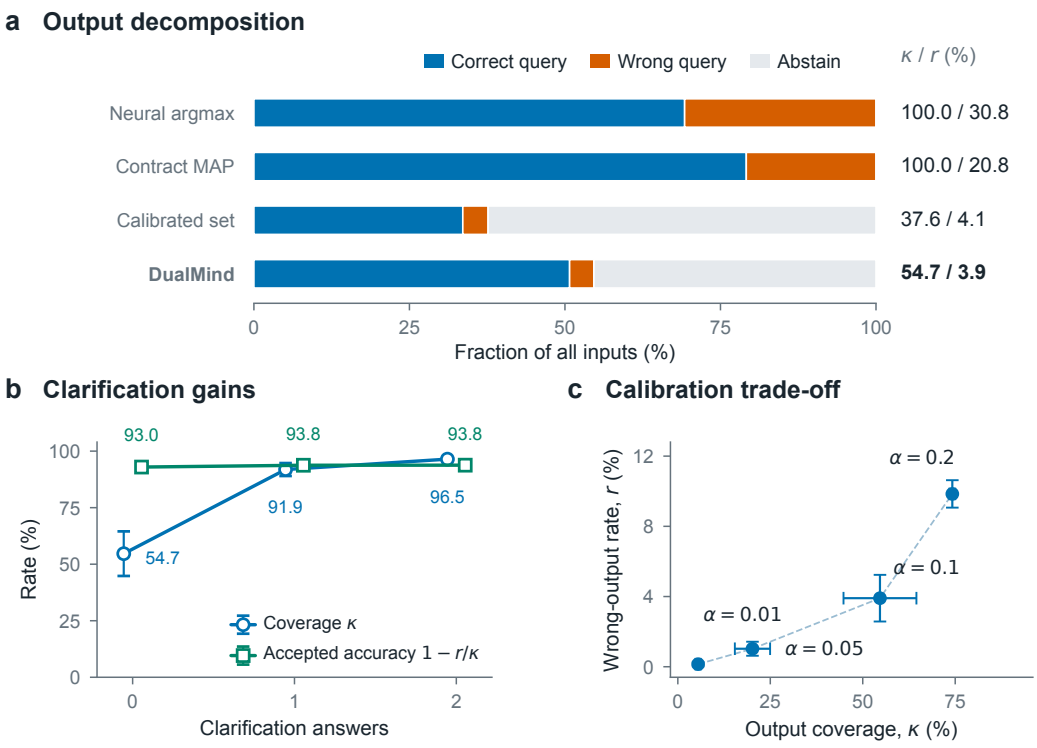


Figure 2: **(a) Output decomposition.** Mean noninteractive correct, wrong, and abstained fractions share the all-input denominator. Labels give coverage $\kappa$ and wrong-output rate $r$. **(b) Clarification gains.** Coverage and accepted-query accuracy $a_{\text{sel}} = 1 - r/\kappa$ are shown for zero, one, and two simulated authoritative answers. **(c) Calibration trade-off.** The noninteractive sweep reports coverage against the all-input wrong-output rate. Points and whiskers in (b,c) show means and $\pm 1$ sample standard deviation over three seeds. These synthetic measurements are not per-query guarantees.

ously unavailable. The budget-optimal, edge, entropy, and random policies are close at budget two (Figure 3 in Appendix D). We do not claim a large empirical advantage for the Bellman policy. Its contribution is the explicit utility objective and exact finite-instance optimum under model weights.

Figure 2(c) shows the noninteractive calibration tradeoff: increasing $\alpha$ raises both coverage and wrong-output rate. The curve therefore exposes the cost of accepting more inputs, rather than identifying a uniformly superior setting. Its error bars summarize variation across three seeds, not per-query guarantees.

**Verification and boundaries.** Executable audits check 27,520 state/question-set configurations against the separation criterion and 676 one-step planning optima, in addition to SQL/interpreter comparisons. These are finite implementation checks, not machine-checked proofs of all theorems. Appendix D reports missing evidence, corrupted evidence, and a clue-distribution shift. Top-1 truncation increases wrong outputs to 19.1%, directly exhibiting the failure mode behind the truncation counterexample. Raising misleading clues to 0.20 without recalibration yields 16.8% wrong outputs. Exchangeability no longer holds. All measurements and model checkpoints are included, together with the unsuccessful and abstained cases.

## 6 Related Work and Positioning

DualMind draws on SQL grounding, neuro-symbolic inference, and uncertainty-aware clarification. Having defined the semantic quotient and one-envelope refinement, we can now state the boundary precisely: the central question is when available evidence justifies one query despite ambiguity among internal interpretations.

**Schema grounding and text-to-SQL.** RAT-SQL models relations and schema linking (Wang et al., 2020). PICARD rejects inadmissible decoding steps using incremental parsing (Scholak et al., 2021). DIN-SQL decomposes the task and applies self-correction (Pourreza & Rafiei, 2023). AmbiQT explicitly studies lexical and structural ambiguity and diversifies plausible SQLs (Bhaskar et al., 2023). DualMind does not claim to discover schema ambiguity. It studies the emission criterion when several candidate interpretations remain, including when different groundings provably have the same semantics. BIRD and Spider 2.0 establish the importance of database contents, external knowledge, and realistic enterprise workflows (Li et al., 2023; Lei et al., 2025). Our controlled study does not substitute for evaluation on those benchmarks. Test-suite evaluation (Zhong et al., 2020) motivates using multiple instances and distinguishing empirical execution agreement from universal equivalence.

**Neuro-symbolic inference.** Scallop integrates probabilistic deductive databases with differentiable reasoning (Huang et al., 2021). Semantic probabilistic layers restrict predictions to symbolic constraints (Ahmed et al., 2022). Thus combining probabilities and logic is established. DualMind focuses on a different interface question: a grounding may satisfy available constraints and still be one of several semantically incompatible interpretations. Certified constraint satisfaction and uncertainty about intent must be represented simultaneously.

**Calibration and questions.** Conformal classification provides marginal label coverage (Romano et al., 2020), and conformal risk control extends calibration to monotone losses (Angelopoulos et al., 2024). KnowNo combines conformal uncertainty with requests for help in language-model planning (Ren et al., 2023). Equivalence-class active learning studies questions that distinguish decision classes (Golovin et al., 2010). DualMind does not present these ingredients as new. It groups candidates by SQL meaning, preserves one calibrated set through evidence refinement, and distinguishes errors over all inputs from errors among accepted queries. These choices make the composition precise and testable.

## 7 Limitations and Conclusion

The framework assumes an enumerated interpretation language and a trusted semantics interface. It cannot recover business meaning absent from both the request and external evidence. Exact planning is exponential in the general finite universe, and the experiment uses synthetic requests and oracle answers rather than real users or a large pretrained model. Within this scope, DualMind provides a concrete bridge from probabilistic schema preference to evidence-based semantic agreement. Its practical message is to preserve plausible alternatives, ask only when their query semantics disagree, and attach guarantees to the complete decision procedure rather than to a likelihood score alone.

## AI USE STATEMENT

Generative AI tools were used to improve the readability of the manuscript, and to assist in the preparation and refinement of scientific figures. All AI-assisted figures were reviewed and checked against the underlying experimental results. The authors take full responsibility for the final manuscript.

## REPRODUCIBILITY STATEMENT

Appendix A provides complete proofs, Appendix B specifies the SQL fragment, and Appendix C describes the data and training protocol. The accompanying archive contains source code, model checkpoints, per-example inputs and labels, calibration and test probabilities, raw metrics, and scripts that regenerate the tables and figures. All data are synthetic and contain no private database records. The software checks and their counts are reported separately from mathematical proofs.

## REFERENCES

Kareem Ahmed, Stefano Teso, Kai-Wei Chang, Guy Van den Broeck, and Antonio Vergari. Semantic probabilistic layers for neuro-symbolic learning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 29944–29959, 2022. doi: 10.52202/068431-2171. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/c182ec594f38926b7fcb827635b9a8f4-Abstract-Conference.html.

Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/f3549ef9b5ff520a7e41ff3cc306ab2b-Abstract-Conference.html.

Adithya Bhaskar, Tushar Tomar, Ashutosh Sathe, and Sunita Sarawagi. Benchmarking and improving text-to-SQL generation under ambiguity. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7053–7074, 2023. doi: 10.18653/v1/2023.emnlp-main.436. URL https://aclanthology.org/2023.emnlp-main.436/.

Daniel Golovin, Andreas Krause, and Debajyoti Ray. Near-optimal bayesian active learning with noisy observations. In *Advances in Neural Information Processing Systems*, volume 23, pp. 766–774, 2010. URL https://proceedings.neurips.cc/paper/2010/file/1e6e0a04d20f50967c64dac2d639a577-Paper.pdf.

Jiani Huang, Ziyang Li, Binghong Chen, Karan Samel, Mayur Naik, Le Song, and Xujie Si. Scallop: From probabilistic deductive databases to scalable differentiable reasoning. In *Advances in Neural Information Processing Systems*, volume 34, pp. 25134–25145, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/d367eef13f90793bd8121e2f675f0dc2-Abstract.html.

Fangyu Lei, Jixuan Chen, Yuxiao Ye, Ruisheng Cao, Dongchan Shin, Hongjin Su, Zhaoqing Suo, Hongcheng Gao, Wenjing Hu, Pengcheng Yin, Victor Zhong, Caiming Xiong, Ruoxi Sun, Qian Liu, Sida I. Wang, and Tao Yu. Spider 2.0: Evaluating language models on real-world enterprise text-to-SQL workflows. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/46c10f6c8ea5aa6f267bcdabcb123f97-Abstract-Conference.html.

Jinyang Li, Binyuan Hui, Ge Qu, Jiaxi Yang, Binhua Li, Bowen Li, Bailin Wang, Bowen Qin, Ruiying Geng, Nan Huo, Xuanhe Zhou, Chenhao Ma, Guoliang Li, Kevin C. C. Chang, Fei Huang, Reynold Cheng, and Yongbin Li. Can LLM already serve as a database interface? A BIg bench for large-scale database grounded text-to-SQLs. In *Advances in Neural Information Processing Systems*, volume 36, pp. 42330–42357, 2023. doi: 10.52202/075280-1835. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/83fc8fab1710363050bbd1d4b8cc0021-Abstract-Datasets_and_Benchmarks.html.

Mohammadreza Pourreza and Davood Rafiei. DIN-SQL: Decomposed in-context learning of text-to-SQL with self-correction. In *Advances in Neural Information Processing Systems*, volume 36,

pp. 36339–36348, 2023. doi: 10.52202/075280-1577. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/72223cc66f63ca1aa59edaec1b3670e6-Abstract-Conference.html.

Allen Z. Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, Zhenjia Xu, Dorsa Sadigh, Andy Zeng, and Anirudha Majumdar. Robots that ask for help: Uncertainty alignment for large language model planners. In *Proceedings of the 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 661–682, 2023. URL https://proceedings.mlr.press/v229/ren23a.html.

Yaniv Romano, Matteo Sesia, and Emmanuel Candès. Classification with valid and adaptive coverage. In *Advances in Neural Information Processing Systems*, volume 33, pp. 3581–3591, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/244edd7e85dc81602b7615cd705545f5-Abstract.html.

Torsten Scholak, Nathan Schucher, and Dzmitry Bahdanau. PICARD: Parsing incrementally for constrained auto-regressive decoding from language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 9895–9901, 2021. doi: 10.18653/v1/2021.emnlp-main.779. URL https://aclanthology.org/2021.emnlp-main.779/.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Bailin Wang, Richard Shin, Xiaodong Liu, Oleksandr Polozov, and Matthew Richardson. RAT-SQL: Relation-aware schema encoding and linking for text-to-SQL parsers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7567–7578, 2020. doi: 10.18653/v1/2020.acl-main.677. URL https://aclanthology.org/2020.acl-main.677/.

Ruiqi Zhong, Tao Yu, and Dan Klein. Semantic evaluation for text-to-SQL with distilled test suites. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 396–411, 2020. doi: 10.18653/v1/2020.emnlp-main.29. URL https://aclanthology.org/2020.emnlp-main.29/.

# A Complete proofs and assumptions

## A.1 Objects, events, and guarantee scope

For each input $x$, $\mathcal{H}(x)$ is a finite set of interpretations and $h^\star \in \mathcal{H}(x)$ is the intended label in the modeled task. This assumption is substantive: a candidate generator cannot justify excluding an unmodeled business interpretation. The ground-truth semantics are $[\![c_x(h^\star)]\!]$, conditional on the representation and compiler. The distribution of $(X, H^\star)$ describes both user requests and their latent intended interpretations. Ambiguous surface requests may therefore have more than one possible label. Ground truth in the synthetic study is sampled before the request is rendered.

The relation $\equiv_x$ is reflexive, symmetric, and transitive because it is pointwise equality of denotations on $\mathcal{D}(x)$. A signature is a map into a discrete set of normal forms. Soundness means

$$\sigma_x(h) = \sigma_x(u) \implies c_x(h) \equiv_x c_x(u). \tag{10}$$

Exactness adds the reverse implication. None of the main results assumes that numerical proximity of embeddings implies logical entailment. The model's probability distribution is allowed to be misspecified. Fixed positive weights are needed only to make the planning ratios well defined. The safety proof does not depend on them.

The event $G$ concerns *all* evidence used before emission or abstention on the actual adaptive path. The event $K$ concerns correctness of the compiler, certified signature equalities, and the declared semantic interface. One can take $\epsilon = 0$ in a mathematical model with a proved compiler/signature. Finite tests cannot establish that an arbitrary software implementation has zero error. The parameters $\delta$ and $\epsilon$ must not be estimated as zero merely because no failures were observed on a small sample.

## A.2 Proof of Proposition 1

Suppose $S \neq \varnothing$, $h^\star \in S$, and $\Sigma(S) = \{z\}$. Select any $\widehat{h} \in S$. Then $\sigma_x(\widehat{h}) = z = \sigma_x(h^\star)$, and soundness implies $c_x(\widehat{h}) \equiv_x c_x(h^\star)$.

For necessity under exactness, suppose $|\Sigma(S)| > 1$. There are $h, u \in S$ with distinct signatures, so exactness implies $c_x(h) \not\equiv_x c_x(u)$. If some query $q$ were correct for both, then $q \equiv_x c_x(h)$ and $q \equiv_x c_x(u)$. Symmetry and transitivity would imply $c_x(h) \equiv_x c_x(u)$, a contradiction. This argument allows $q$ outside the compiler image and thus is not merely a statement about which candidate is selected.

A randomized single-query rule with probability-one correctness for every member of a finite $S$ cannot evade the argument. For each $h \in S$, the random seeds producing a correct query have probability one. The finite intersection of these seed events also has probability one. A seed in that intersection would produce a query correct for every member, which is impossible when two members disagree. Randomization with nonzero allowed error is a different decision criterion.

## A.3 Split-conformal coverage with ties

Condition on the independently fitted model. The calibration scores $Z_i = 1 - p_\theta(H_i^\star \mid X_i)$, $i \leq n$, and test score $Z_{n+1}$ are exchangeable. Let $k = \lceil (n+1)(1-\alpha) \rceil$. If $k = n+1$, the threshold is $+\infty$ and coverage is one. Otherwise, break all score ties with independent continuous random variables for a rank argument. Exchangeability implies that the randomized test rank is uniform on $\{1, \ldots, n+1\}$. A rank at most $k$ implies that the test score is no larger than the $k$th calibration score. Retaining ties with $\leq$ can only increase coverage. Hence

$$\mathbb{P}\{Z_{n+1} \leq Z^{\text{cal}}_{(k)}\} \geq \frac{k}{n+1} \geq 1 - \alpha. \tag{11}$$

This is a marginal statement over calibration and test randomness, conditional on training. It is not a guarantee for every realized calibration set, every request, or every subgroup. A single seed's test miscoverage can exceed $\alpha$ without contradicting it. The score is a simple probability-threshold score. It is not the randomized adaptive prediction-set score proposed by Romano et al. (2020).

If a system first retrieves a reduced candidate universe, the coverage argument applies only to a calibration procedure whose label space and score handle that retrieval stage correctly. We enumerate

all 16 worlds and do not estimate a retrieval recall correction. The more general $\eta$ formulation in Theorem 2 can accommodate any valid bound on initial-set failure, including a separately justified approximation term.

### A.4 Proof of Theorem 2

Define $F = \{H^\star \notin C(X)\}$. On $F^c \cap G$, initial evidence preserves $H^\star$, so $H^\star \in S_0$. If $H^\star \in S_t$, then truth preservation of the next answer gives $H^\star \in E_{t+1}$ and thus $H^\star \in S_t \cap E_{t+1} = S_{t+1}$. This is a pathwise induction. It does not condition on an independently chosen question, so questions may depend arbitrarily on scores and previous answers.

Consequently, on $F^c \cap G \cap K$, any time $\tau$ at which a nonempty consensus set is emitted contains the true interpretation and has a sound signature. Proposition 1 gives correctness. If the policy never emits, it contributes no wrong output. Thus the event $W$ of a wrong emission satisfies

$$W \subseteq F \cup G^c \cup K^c, \qquad \mathbb{P}(W) \leq \eta + \delta + \epsilon. \tag{12}$$

The probability cannot exceed one. When $\kappa > 0$, divide by $\kappa$ using the definition of conditional probability. The conformal result substitutes the previous subsection's $\eta \leq \alpha$. No independence between the three failure events is used. No error budget is spent again merely by inspecting a truth-preserving subset of the same envelope.

If an additional pruning operation can remove $H^\star$ on an event $P$ with probability at most $\beta$, add $P$ to the union. If pruning failure is bounded only conditional on initial inclusion, its unconditional probability is still at most the same $\beta$, so the union-bound expression remains valid, possibly loose. If the system instead regenerates a new set with an uncalibrated rule, there is no reason that a bound for the original $F$ controls its new failures.

**Why accepted-query error needs a denominator.** Consider a procedure that accepts only on a subset of initial-set failures, with probability $\eta > 0$, and is wrong on every such acceptance. It has $r = \eta$, $\kappa = \eta$, and $r_{\text{sel}} = 1$. A joint error bound by $\eta$ therefore cannot imply accepted-query error at most $\eta$. This example explains why both denominators are reported in the main table.

### A.5 Proof of Theorem 3

For a fixed question collection $A$, define the answer vector $g_A(h) = (g_a(h))_{a \in A}$. The possible nonempty transcript cells are $S_z = \{h \in S : g_A(h) = z\}$. If Eq. (8) holds and a cell contained $h, u$ with distinct semantic signatures, a separating question would force their answer vectors to differ. Hence every cell is homogeneous. Conversely, if Eq. (8) fails for some disagreement pair, both members occupy the same cell, which is not homogeneous. This proves equivalence.

For the adaptive impossibility claim, take $h, u$ with different signatures and equal answers to every available question. Couple the random seed of any randomized policy across these two worlds. Initially their observable information and seed are the same. At each step, the policy chooses the same question. Both worlds return the same answer. Induction yields identical complete transcripts and outputs. No single emitted query can be correct in both worlds by Proposition 1. Therefore the policy cannot be uniformly correct and always emit. This claim concerns zero-error identification without additional information. A nonzero-error Bayesian guess remains possible.

If an available separating collection $A$ exists, asking its questions and intersecting the answer cells leaves a homogeneous set containing the true world, provided it was initially retained. This gives the sufficiency statement. The minimum-cost nonadaptive collection is exactly

$$\min_{A' \subseteq A} \sum_{a \in A'} c_a \quad \text{subject to} \quad \bigcup_{a \in A'} \{\{h, u\} \in U(S) : g_a(h) \neq g_a(u)\} = U(S). \tag{13}$$

This is a representation as weighted set cover, not a novel solver for set cover. An adaptive policy may cost less on some paths because it need not identify all worlds simultaneously.

### A.6 Proof of Theorem 4

We use induction on $b$. An empty state cannot emit. A homogeneous nonempty state can emit immediately with value one. An unresolved state with $b = 0$ cannot emit under the consensus rule. These establish the boundary cases.

For $b \geq 1$, any deterministic policy at an unresolved state either abstains, with value zero, or asks a first question $a$. Under the assumed weights, answer $o$ occurs with probability $w_S(S_{a,o})$. Conditional weights in the child are

$$\frac{w_S(h)}{w_S(S_{a,o})} = \frac{p_\theta(h \mid x)}{\sum_{u \in S_{a,o}} p_\theta(u \mid x)} = w_{S_{a,o}}(h). \tag{14}$$

Thus the induction hypothesis applies to each child with budget $b-1$. Summing bounds the policy's value by the expression for that $a$ in Eq. (6). Maximization bounds every possible first question. Conversely, choose a maximizing first question and an attaining child policy, which exists by finiteness and induction. Their composition attains the recurrence. Randomized policies give convex combinations of first-action values, and cannot exceed their maximum.

Questions with a single nonempty answer cell can be skipped: they reveal nothing and consume budget. Repeating a deterministic question also reveals nothing. For four binary attributes, each attribute along a path is either unasked, fixed to zero, or fixed to one. There are at most $3^4 = 81$ distinct restricted states for a fixed initial set. The general $2^N$ subset bound is deliberately loose. The reported policy optimizes success probability within a cap. Among tied optima, the implementation uses the first coordinate. It does not additionally minimize the expected number of answers, and is not guaranteed to outperform an edge heuristic on that secondary metric.

## B SQL language, compiler, and semantic proof

### B.1 Schema and legal instances

An order row has a unique integer ID, customer ID, order-day integer, integer billed amount, and binary completion status. A payment row has a unique integer ID, an existing order ID, received-day integer, integer paid amount, and binary completion status. Amounts are nonnegative integer cents. There are no nulls. A legal instance is finite, satisfies primary and foreign keys, and keeps aggregate values within the SQLite integer range. This last restriction avoids treating mathematical integer sums and overflowing implementation operations as identical. No floating-point money, refunds, outer joins, nested subqueries, or arbitrary predicates are modeled.

Let $R_h(D)$ be the selected multiset of records. For $m = 0$, it contains each order whose day lies in $[30, 60)$ and, if $s = 1$, whose status is one. The value of $t$ is irrelevant. For $m = 1$, it contains each payment joined to its unique order, filtered using received day when $t = 0$ or order day when $t = 1$, and filtered on payment status if $s = 1$. The join contains exactly one row per payment before filtering. A query returns $|R_h(D)|$ for $a = 1$, and the sum of its source-specific amount values for $a = 0$, with an empty sum equal to zero.

### B.2 Compiler examples

For $h = (1, 0, 1, 0)$, the intended meaning is completed payment amount by received date:

```
SELECT COALESCE(SUM(p.paid_cents), 0)
FROM payments p JOIN orders o ON p.order_id = o.id
WHERE p.received_day >= 30 AND p.received_day < 60
  AND p.status = 1
```

For $h = (0, 1, 0, 0)$, the compiler instead uses the billed amount on orders:

```
SELECT COALESCE(SUM(o.billed_cents), 0)
FROM orders o
WHERE o.ordered_day >= 30 AND o.ordered_day < 60
```

The latter is identical to the query for $(0, 0, 0, 0)$. The compiler selects only known identifiers and the declared fixed predicate. It is not a parser or verifier for arbitrary model-written SQL. Prefix-renamed table identifiers in the implementation audit test compiler portability under consistent renaming, not learned schema generalization.

### B.3 Compiler soundness and exact signatures

**Proposition 5.** *On the legal instances just defined, the compiler implements $R_h$ and its aggregation. Further, $c(h) \equiv c(u)$ if and only if $\sigma(h) = \sigma(u)$ for Eq. (9).*

*Proof.* For order-source queries, the `FROM` clause enumerates each order exactly once. The `WHERE` clause expresses the two bounds on its day and the optional status test. For payment-source queries, the foreign key and unique referenced key produce exactly one joined row per payment. The selected day and status expressions implement the specified filtering. Since values are non-null and no overflow occurs, `COUNT(*)` computes multiset cardinality and `SUM` computes the sum of amounts. `COALESCE` maps the empty-sum null to zero, agreeing with the definition. This proves the compiler statement.

Equal signatures make all relevant source, filter, date, and aggregation choices equal. The only possible differing bit is $t$ with $m = 0$, which the compiler ignores. Therefore the queries are equivalent.

For the reverse direction, we give separating legal databases for any pair with different signatures. If their source bits differ, use one completed order at day 40 with billed amount 2, and no payments. Every order query returns either 1 or 2, while every payment query returns zero, independently of the other bits. If sources agree but aggregation bits differ, use one completed order and, for a payment source, one completed payment referring to it. Set both dates to 40 and both amounts to 2. The sum is 2 and the count is 1, irrespective of date and status choices. If source and aggregation agree but status bits differ, use a single relevant source record of status zero, dates 40, and amount 2 (plus its required order when the source is payments). The all-record query is positive and the completed-record query is zero. Finally, if only the effective date bit differs, the source must be payments. Use one completed order at day 40 and one completed payment at received day 10, with amount 2. The order-date query is positive and the received-date query is zero. These cases exhaust distinct signatures and prove exactness. □

### B.4 Implementation audit

The audit constructs 256 databases using seed 871. The first is empty. Each remaining database has between 1 and 14 orders and between 0 and 24 payments. Foreign keys are generated by selecting an existing order. Days are drawn from $\{0, \ldots, 89\}$, amounts from $\{1, \ldots, 9999\}$, and status from $\{0, 1\}$. All 16 queries are executed on each database. A separate loop-based Python evaluator implements the row-selection definition without calling the compiler or signature function. All 4,096 scalar outputs agree. The 16 output vectors across the suite agree if and only if their signatures agree. This is a finite check of the implementation, while Proposition 5 gives the universal claim.

## C Experimental protocol and reproducibility

### C.1 Generation, model, and calibration

The sampled world is uniform on $\{0, 1\}^4$. Each of four labeled slots is rendered using one of three synonyms for its sampled bit. Slot order and two context words are randomized. An omitted clue is rendered as `unspecified`. A present clue is independently changed to the opposite bit's synonym with probability $0.03$. The syntax is deliberately simple so that a small neural component can be trained without external model downloads. This construction studies erroneous or incomplete grounding evidence. It does not reproduce all sources of real LLM error.

For example, a generated input may contain "source received, status all, aggregate total, date unspecified." The model sees a tokenized, templated request, not gold bits, metadata masks, or clarification

answers. The static catalog-to-plan mapping is supplied by the compiler. Thus the experiment cannot establish learned alignment to unseen table or column descriptions. The task predicts a single categorical plan token with causal attention over the prefix, not a freely generated SQL string.

The vocabulary has the context, slot, synonym, and special tokens defined in `code/experiment.py`. Sequences have length 15, including one trailing padding position. Prediction uses the `PLAN` position at index 13, which cannot attend to the future padding position. Two pre-normalized Transformer layers have four attention heads, width 48, feed-forward width 96, and zero dropout. Learned position embeddings and a 16-way output projection give 41,344 parameters. Training uses cross-entropy and AdamW with weight decay $0.01$, learning rate $0.001$, batch size 128, and 12 fixed epochs. All computation uses CPU and four PyTorch threads, with deterministic algorithms enabled.

For seed $s \in \{11, 23, 37\}$, training, calibration, nominal test, shifted test, and metadata generators use seeds $s + 1000$, $s + 2000$, $s + 3000$, $s + 4000$, and $s + 5000$, respectively. Model initialization and training shuffles use $s$. The split sizes are 4,096, 1,024, 2,048, and 2,048. No development set or hyperparameter search was used. Model architecture and settings were fixed before the reported training. The experiment was rerun after adding the exact clarification policy. This did not change the neural architecture, data, or training settings. Identical predictions resulted for the original methods.

### C.2 Evidence and questions

Each attribute is independently revealed by initial certified evidence with probability $0.35$. This is a favorable but explicit assumption about availability of business information. The same reveal mask and values are regenerated for all applicable methods. Evidence filtering compares candidate bits to revealed values. Missing values impose no restriction. The corruption stress test independently flips each revealed value with probability $0.10$. The model is not given these masks as input. Contract-filtered and DualMind methods both have access to them after prediction.

There are four possible clarification questions, one per attribute. The oracle returns the sampled true bit. These answers are additional information, not free corrections inferred from a verifier. A method asks only while the retained set is nonempty and semantically unresolved. Budget counts the maximum number of answers per input. The reported average answer count includes accepted-without-question and abstained cases. Because some initial envelopes exclude the truth, asking questions may make a set empty rather than recover a correct plan. We do not restart with a new uncalibrated envelope.

The exact policy optimizes model-weighted coverage. The edge baseline computes expected removal of cross-semantic pair mass. The entropy baseline asks the bit with highest binary entropy under normalized retained model weights. Random selection is uniform among coordinates with two nonempty answer cells. Tie breaking in deterministic policies uses coordinate order. Edge and entropy weights are recomputed by normalization on the surviving set, but initial conformal membership is never changed.

### C.3 Metrics and uncertainty reporting

An accepted plan is correct when its signature equals the sampled world's signature. This criterion uses the proved exact fragment semantics. The compiler receives a separate implementation audit. Table 1 reports the arithmetic mean and sample standard deviation of each seed's metric. These standard deviations describe three independently trained synthetic runs. They are not confidence intervals or significance tests. In particular, averaged selective error is the mean of the three ratios, not a ratio silently substituted for those values.

`results/cases_*.jsonl` contains nominal, shifted, and calibration input strings, labels, and compiled reference SQL. `predictions_*.npz` contains the full 16-class probability arrays and labels. Metadata and question traces can be regenerated deterministically from the documented seeds. `run_*.json` contains all metric settings and measured elapsed times. `code/summarize.py` generates every table and numeric macro from these records. No model API, private data, or paid service is required.

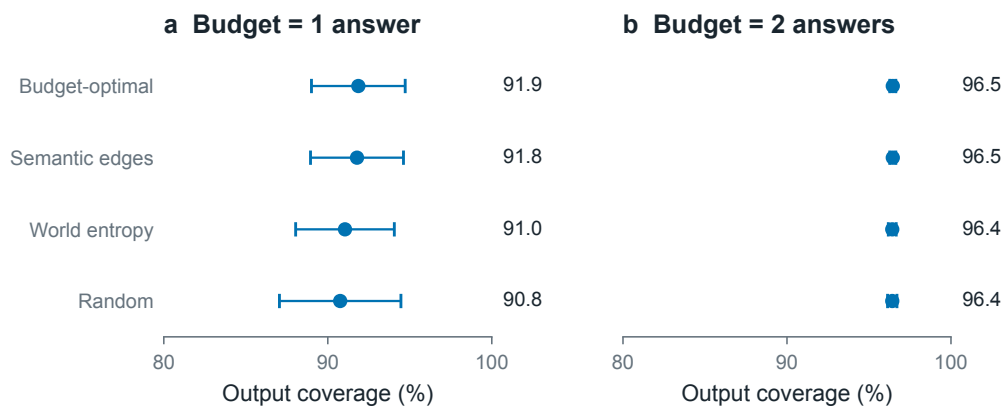


Figure 3: Clarification policy comparison at matched answer caps. Points and whiskers show mean coverage and $\pm 1$ sample standard deviation over three seeds. The horizontal axes span 80–100%. The policies have similar coverage at budget two. Model-weighted optimality does not imply empirical dominance.

**Local resource footprint.** The experiments ran on a 16 GB Apple M4 using CPU only, Python 3.10.19, PyTorch 2.13.0, NumPy 2.2.6, and SQLite 3.51.0, as recorded by the environment. Measured training averaged approximately 2.4 seconds per tiny model, and nominal inference for 2,048 requests averaged approximately 0.059 seconds. These short microbenchmark timings describe this 41k-parameter diagnostic and exclude interpreter startup, rendering, and writing the paper. They do not estimate the cost of deploying a pretrained language model. No memory profiling claim is made.

## D Additional results and assumption violations

Table 2: Clarification policies at matched maximum answer budgets. Percentages are mean $\pm$ sample standard deviation across three seeds. Answer counts are means over all inputs. "Optimal" refers to the model-weighted Bellman objective, not true-distribution optimality or minimum expected answer count.

| Policy | Coverage | Wrong outputs | Answers |
|---|---|---|---|
| Optimal, budget 1 | $91.9 \pm 2.9$ | $5.7 \pm 1.1$ | 0.419 |
| Edge, budget 1 | $91.8 \pm 2.8$ | $5.7 \pm 1.1$ | 0.419 |
| Entropy, budget 1 | $91.0 \pm 3.0$ | $5.8 \pm 1.1$ | 0.419 |
| Random, budget 1 | $90.8 \pm 3.7$ | $5.7 \pm 1.1$ | 0.419 |
| Optimal, budget 2 | $96.5 \pm 0.2$ | $6.0 \pm 1.1$ | 0.482 |
| Edge, budget 2 | $96.5 \pm 0.2$ | $6.0 \pm 1.0$ | 0.468 |
| Entropy, budget 2 | $96.4 \pm 0.2$ | $6.0 \pm 1.0$ | 0.475 |
| Random, budget 2 | $96.4 \pm 0.3$ | $6.0 \pm 1.0$ | 0.478 |

Figure 3 and Table 2 show that the four policies have similar coverage at budget two. The exact policy does not dominate the heuristics on all measured quantities, particularly mean answer count. This is consistent with its objective: a cap on answers with maximum model-weighted resolution probability, rather than a penalty on every question. Distinguishing those objectives avoids turning an optimality theorem into a stronger empirical claim than it supports.

Smaller $\alpha$ retains more interpretations and leads to more abstention, while larger $\alpha$ accepts more inputs. Test retention is not exactly $1 - \alpha$ in every seed. The guarantee averages over calibration and test draws, and the finite experiment also has sampling variation. The fixed confidence-0.9 baseline has no calibrated error guarantee and is included only as an interpretable comparator.

With all four attributes correctly revealed, every surviving set identifies the true world and no wrong outputs occur. Coverage remains below one because an initial calibrated set may omit that world.

Table 3: Noninteractive calibration sweep. Retention is the fraction of true worlds in the initial evidence-filtered set. With truthful evidence it equals calibrated-set coverage on the test sample. Other columns use the definitions in Eq. (2).

| $\alpha$ | Retention | Coverage | Wrong outputs | Selective error |
|---|---|---|---|---|
| 0.01 | $99.2 \pm 0.3$ | $5.5 \pm 0.7$ | $0.1 \pm 0.1$ | $2.6 \pm 2.2$ |
| 0.05 | $94.8 \pm 1.0$ | $20.2 \pm 4.8$ | $1.0 \pm 0.4$ | $5.0 \pm 0.8$ |
| 0.10 | $89.8 \pm 1.2$ | $54.7 \pm 9.9$ | $3.9 \pm 1.3$ | $7.0 \pm 1.4$ |
| 0.20 | $79.7 \pm 0.9$ | $74.2 \pm 0.8$ | $9.8 \pm 0.8$ | $13.3 \pm 1.1$ |

Table 4: Evidence and approximation stress tests, noninteractive DualMind at $\alpha = 0.1$. Percentages are mean $\pm$ sample standard deviation. "Revealed bits" are externally supplied evidence. Corrupted evidence and shifted clues violate nominal assumptions.

| Condition | Retention | Coverage | Wrong outputs |
|---|---|---|---|
| No certified attributes | $89.8 \pm 1.2$ | $37.6 \pm 11.5$ | $4.1 \pm 1.1$ |
| 35% certified attributes | $89.8 \pm 1.2$ | $54.7 \pm 9.9$ | $3.9 \pm 1.3$ |
| 70% certified attributes | $89.8 \pm 1.2$ | $73.6 \pm 5.7$ | $2.4 \pm 0.6$ |
| All attributes certified | $89.8 \pm 1.2$ | $89.8 \pm 1.2$ | $0.0 \pm 0.0$ |
| Top-1 truncation | $66.0 \pm 1.5$ | $87.2 \pm 0.6$ | $19.1 \pm 1.4$ |
| Top-2 truncation | $85.4 \pm 0.8$ | $57.4 \pm 8.7$ | $5.0 \pm 0.8$ |
| Top-4 truncation | $89.6 \pm 1.1$ | $54.7 \pm 9.8$ | $3.9 \pm 1.3$ |
| 10% corrupted revealed bits | $77.9 \pm 1.1$ | $48.8 \pm 7.7$ | $5.9 \pm 0.4$ |
| 20% misleading clues (shift) | $49.6 \pm 1.3$ | $45.5 \pm 6.8$ | $16.8 \pm 2.4$ |

Subsequent intersections cannot add it back. This is a deliberate property of the one-envelope method, not a claim that a production system should never support a separately justified recovery path.

Top-$k$ truncation is applied after conformal selection and before evidence intersection. At $k = 1$ it often converts ambiguity into unsupported confidence, raising both coverage and wrong outputs. At $k = 4$ the effect is small because most relevant nominal sets are already small. That empirical observation does not justify arbitrary truncation in general.

The shifted test distribution changes the clue-flip probability from $0.03$ to $0.20$, retaining the original calibration threshold. Its elevated wrong-output rate is outside the exchangeable guarantee. The corruption test changes evidence, so the truth-preservation event may fail. These tests locate the method's dependence on data and metadata quality. They do not show that formal reasoning repairs false premises.

## E Reproduction commands and certificate audit

From the archive root, with NumPy, PyTorch, and Matplotlib installed:

```
python code/experiment.py --out results --seeds 11 23 37
python code/verify.py
python code/summarize.py
```

The experiment writes checkpoints, full probability arrays, cases, raw metrics, and environment versions. The verification script checks the separation criterion for 27,520 state/question-set configurations, truth-preserving refinement on their retained worlds, and 676 one-step planning choices against a directly enumerated objective. It then reruns the independent SQL audit. These are executable checks of finite instances, not Lean or SMT proof certificates. The mathematical arguments are the proofs in this appendix.